Angular introduces two decorators: @Input and @Output. They are used to communicate between components and trigger events. We’ll look at how to pass data to the components using @Input and to get the data back flowing with @Output.

To understand how data is passed to any component we can see the code example below:

Graphical user interface, text, application, email

Description automatically generated

(Chintala, 2017)

So, parentCount is an @Input binding for the child element and fullName is the local component’s property.

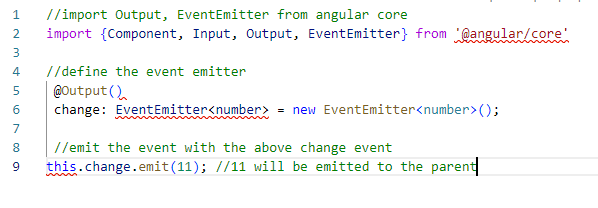
To obtain data passed from a component we use the @Input decorator. To capture the data in the child component coming from the parent we will again use the @Input. Let’s look at the code below:

Text

Description automatically generated

(Chintala, 2017)

The above code examples show how to pass data from the parent to the child component. Lets look at how to connection between parent and child. We’ll be using the @Output decorator and an EventEmitter. We do this so that we can “emit the event to the parent so that parent can capture the exposed event and retrieve data.” (Chintala, 2017) We need to decorate the child component with @Input, which will pass initial data to the child component and get back updated data using @Output, and Event emitter. The code would look something like this:



(Chintala, 2017)

To let the parent know that a change has taken place we can use the sample code below and put everything together: 

Chintala (2017)

As we can see from the descriptions and code structures above we can construct very dynamic and responsive applications for our organizations and the end users that we are ultimately building for. A good example of how we could use @Input and @Output with EventEmitters would be to show a list of user names on an application and then when a user name is clicked details of that particular users are displayed.

References:

Chintala, K. (2017, August 10). *@Input and @Output in Angular*. CodeProject. Retrieved February 8, 2023. From <https://www.codeproject.com/Articles/1200763/Input-and-Output-in-Angular>